1. Download a free build of the software (the official site version requires a license, but it's GPL-ed, so you can find free binaries such as this one, which is a simple and functional installer for Windows);
2. Generate your traditional HTML docs in your preferred way (Ant, Maven, command-line javadoc - your choice);
3. The GUI is nice, but adding all files manually can be cumbersome, so just create a .book file with Document Type set to "Web Page", add one of the HTML files from your generated javadoc's root folder (e.g., overview-summary.html, anyone will do, it's just for reference on step 5). On the Output tab select the PDF format and set a name for it, and add other options to your heart's content (logos, colors, lots of cool stuff here). Save this project (say, `myjavadocpdf.book`) and close the GUI
4. Generate a list of all HTML files in your javadoc. I did it with Cygwin's find command (my DOS/cmd shell days are long over), but you can do anything you want, as long as you get a file list. In my case a `find . | grep html$ | sort -r > files.txt` did the trick; For all windows users `dir /s/b *.html > files.txt` should do the same.
5. Open the .book file generated on step 3 in your favorite pure text editor (as a programmer you should have strong opinions on that, so will keep my opinions to myself - NOT ;-)) and append the list generated on step 4 to this .book file (it keeps the list of files at the end, making life really easy). Don't forget to fix the relative paths, if needed with a global search/replace (that's why you needed at least one file added on step 3 - to see which file path pattern htmldoc expects);
6. Now you should sort the files in a convenient order. I put my overview first, then package descriptions and each class, then the full index, and everything else at the end. Remember that any file you delete will become an external (hence broken) link, so choose wisely;
7. Save your .book file and re-open it on HTMLDOC. Voila: all files added and sorted. Click on generate. That's it!